



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA PODNIKATELSKÁ

FACULTY OF BUSINESS AND MANAGEMENT

ÚSTAV INFORMATIKY

INSTITUTE OF INFORMATICS

VÝVOJ MOBILNÍ APLIKACE PRO PLATFORMU IOS

DEVELOPMENT OF MOBILE APPLICATION FOR IOS PLATFORM

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Ondřej Andrýsek

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Petr Dydowicz, Ph.D.

BRNO 2017

Abstrakt

Bakalářská práce se zabývá návrhem a vývojem nativní mobilní aplikace pro platformu iOS. Aplikace je vyvíjena v prostředí XCode a napsána v programovacím jazyce Swift. Následně bude aplikace distribuována v obchodě App Store.

Abstract

Bachelor's thesis is based on design and development of native application for platform iOS. Application is developed in XCode and wrote in programming language Swift. Then the application is distributed in App Store.

Klíčové slova

iOS, mobilní aplikace, Swift, App Store, XCode

Key words

iOS, mobile application, Swift, App Store, XCode

Bibliografická citace

ANDRÝSEK, O. *Vývoj mobilní aplikace pro platformu iOS*. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2017. 51 s. Vedoucí diplomové práce Ing. Petr Dydowicz, Ph.D.

Čestné prohlášení

Prohlašuji, že předložená bakalářská práce je původní a zpracoval jsem ji samostatně.
Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil
autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech
souvisejících s právem autorským).

V Brně dne 31. května 2017

.....

podpis studenta

Poděkování

Chtěl bych poděkovat vedoucímu bakalářské práce Ing. Petru Dydowiczovi, Ph.D., za pomoc a rady, které mi poskytl k této bakalářské práci.

Obsah

ÚVOD.....	9
1 CÍL A METODIKA PRÁCE.....	10
2 TEORETICKÁ VÝCHODISKÁ PRÁCE	11
2.1 Mobilní aplikace	11
2.2 iOS	11
2.3 Android.....	12
2.4 Windows Phone	12
2.5 Jazyk Swift 3	12
2.6 Porovnání s Objective – C.....	13
2.7 Objektové programování	14
2.8 App Store.....	14
2.9 XCode	15
2.10 CocoaPods	15
2.11 Knihovny.....	15
2.12 SWOT analýza.....	16
3 Analýza současného stavu.....	17
3.1 Podíl mobilních platforem na trhu	17
3.2 Google Play.....	18
3.3 App Store.....	18
3.4 Monetizační modely.....	19
3.4.1 Zobrazení reklam.....	19
3.4.2 Placená aplikace	20
3.4.3 Nákupy v aplikaci.....	20
3.4.4 Freemium	21
3.4.5 Předplatné	22
3.5 Analýza stávající webové stránky.....	23
3.6 SWOT Analýza	24
Silné stránky	24
Slabé stránky	25
Příležitosti.....	25
Hrozby	25

3.7	Shrnutí	25
4	Vlastní návrh řešení, přínos práce	27
4.1	Návrh tříd	29
4.2	Přihlašovací obrazovka	30
4.3	Registrační obrazovka	31
4.4	Uživatelé v okolí	32
4.5	Detail uživatele	35
4.6	Hodnocení fotek	36
4.7	Menu	37
4.8	Konverzace	38
4.9	Zprávy	39
4.10	Moje fotky a videa	40
4.11	Listy fotek	41
4.12	Nastavení	42
4.13	Ekonomické zhodnocení	43
4.14	Přínos práce	44
	Závěr	45
	Seznam dostupných zdrojů	46
	Seznam obrázků	49
	Seznam tabulek	50

ÚVOD

V dnešní době vlastní většina lidí smartphone, či jiné mobilní zařízení. Tento trend stále stoupá a není divu, že je stále větší poptávka po mobilních aplikacích. Lidé poptávají mobilní hry, health & fitness nebo jiné aplikace, které by jim ulehčili život. I když mnoho podniků se snaží poptávce vyhovět, stále je poptávka vyšší než nabídka. Proto je výhodné vstoupit na trh. Buď jako podnikatelský subjekt nebo jako zaměstnanec podniku. Nabídka po zkušených programátorech v tomto oboru je vysoká. Pro ty méně zkušené vyšel nový programovací jazyk Swift, který se chce zavděčit nováčkům i profesionálům svojí rychlostí a jednoduchostí.

1 CÍL A METODIKA PRÁCE

Zobrazování internetových stránek na mobilu je většinou nepřehledné, text i tlačítka jsou malé a uživateli způsobuje více nepříjemností než potěšení. V lepších případech jsou stránky optimalizované pro mobilní zařízení, stále ale existuje elegantnější řešení. V dnešní době má stále více internetových stránek i svoji vlastní aplikaci, která se snaží maximálně zpříjemnit zážitek v poskytování svých služeb. Takové aplikace bývají přehledné a jednoduché. Ovládání je intuitivní a přesné. Internetové služby se díky aplikaci stávají více konkurenceschopné a mohou být přednostní volbou. Zároveň bude uživatel více využívat vašich služeb.

Gaynet je švýcarská internetová gay seznamka, která má velmi zastaralý web. Seznamka má denně stovky uživatelů online, kteří musejí používat služby pouze prostřednictvím prohlížečů. Pro některé uživatele, kteří chtějí využívat seznamku například na cestě nebo u oběda je to velmi problematické. Proto je mým hlavním cílem této bakalářské práce navrhnout a implementovat aplikaci, která bude jednoduchá a přínosná pro uživatele, kteří chtějí využívat seznamku i mimo počítač. Aplikace ulehčí stovkám uživatelů přístup k informacím a zvýší využívání služeb. Aplikace bude mít moderní design a mohla by přilákat další potenciální uživatele.

V této práci navrhnu schéma logiky obrazovek, kde se bude dát zjistit, kdy se má daná obrazovka objevit. Poté popíšu veškeré obrazovky, které bude potřeba implementovat, aby aplikace zajistila plnohodnotnou náhradu za internetové stránky. Aplikace bude komunikovat se serverem a posílat notifikace o nových událostech.

2 TEORETICKÁ VÝCHODISKÁ PRÁCE

V první části teorie se budu zabývat obecně o mobilních aplikacích a porovnání třech nejznámějších mobilních platforem. Dále se budu věnovat programovacímu jazyku Swift 3 a porovnání s jazykem Objective – C. V další části bude popsáno vývojové prostředí XCode, obchod App Store a knihovny, které budou použity. V poslední části bude zmínka o SWOT analýze.

2.1 Mobilní aplikace

Mobilní aplikace je softwarová aplikace vyvinutá speciálně pro malá zařízení, která jsou bezdrátová, např. smartphony nebo tablety. Mobilní aplikace jsou navrhovány podle požadavků zařízení a mohou využívat výhody, které nabízí. Jedná se například o accelerometer. Mobilní aplikace můžeme rozdělit na webové aplikace, nativní aplikace, které jsou vytvořené speciálně pro jednu platformu a na hybridní aplikace, které kombinují předešlé webové a nativní [1].

2.2 iOS

iOS využívají zařízení společnosti Apple, jako jsou iPhone, iPad, iPod Touch, Apple Watch a Apple TV. iOS je odlehčenou verzí operačního systému MacOS X, mobilní verze má navíc podporu dotykového ovládání. iOS je uzavřený systém, který omezuje možnosti uživatelské instalace aplikací bez využití nástrojů Apple [2], [3].

Aplikace určené pro iOS se distribuují prostřednictvím App Store. Každá aplikace umístěná v App Store musí projít poměrně náročným schvalovacím procesem ze strany Apple, což má jak pozitivní, tak negativní důsledky. Pozitivem je fakt, že na App Store najdeme velmi kvalitní aplikace, které v drtivé většině neobsahují žádný vir ani malware. Aplikace pro iOS se vyvíjí v jazyku C, Objective – C a Swift. Na platformě Mac OS X je k dispozici vývojové prostředí XCode [2], [3].

2.3 Android

Android je nejrozšířenějším mobilním operačním systémem. Je používán v mobilních telefonech, tabletech a navigacích. Jde o otevřenou platformu s otevřeným zdrojovým kódem (open source), založenou na linuxu. Původně byl vyvíjen společností Android, kterou v roce 2005 převzal Google. První verze systému Android 1.0 byla představena v září roku 2008 [2], [3].

Aplikace jsou distribuovány primárně přes Google Play. Zde jsou ke stažení veškeré aplikace Android včetně her. Nahrávání na Google Play je možné přímo z PC a nepodléhá tedy kontrole jako u iOS Apple. Aplikace pro Android se vyvíjí s pomocí Android SDK, s využitím programovacího jazyka Java a knihoven distribuovaných Googlem [2], [3].

2.4 Windows Phone

Operační systém Windows Phone je mobilním systémem vytvářený společností Microsoft. Microsoft zaspal v oblasti operačních systémů, což se odrazilo na počtu aplikací, vyvíjených pro Windows Phone. Distribuce aplikací probíhá přes Windows Phone Marketplace, kde je nedostatek kvalitních aplikací. S dominantním trhem pro Android, který roste ze všech nejrychleji se ani nepředpokládá, že by se o Windows Phone výrazně zvýšil zájem. Výhodou Windows Phone je dobrá interakce se softwarem od Microsoft, a proto se nejvíce využívá ve firemním prostředí [2], [3].

2.5 Jazyk Swift 3

Swift je nový moderní programovací objektivně-orientovaný jazyk pro iOS, OS X, watchOS a tvOS aplikace, který staví na základech C a Objective – C bez nutnosti kompability. Swift byl poprvé představen veřejnosti v roce 2014. Swift adoptoval kontrolu psaní kódu a je podpořen Cocoa a Cocoa Touch frameworky. Pro swift byl vylepšen existující kompilátor, debugger a framework infrastruktura. Byla zjednodušena správa paměti, o kterou se stará ARC (Automatic Reference Counting). Swift se také liší od ostatních jazyků, že se za syntaxí nepíše středník, stejně jako např. u jazyka Python. Novinka od verze 2.1 je nápověda u proměnných, které se nemění a

lze je tedy považovat za konstanty. Tím se mírně zrychlí zpracovávání kódu. Swift také představil optional values, tedy optimální proměnné, které mohou mít hodnotu nebo mohou být nulové (nil), není tedy potřeba null pointerů [4], [5].

2.6 Porovnání s Objective – C

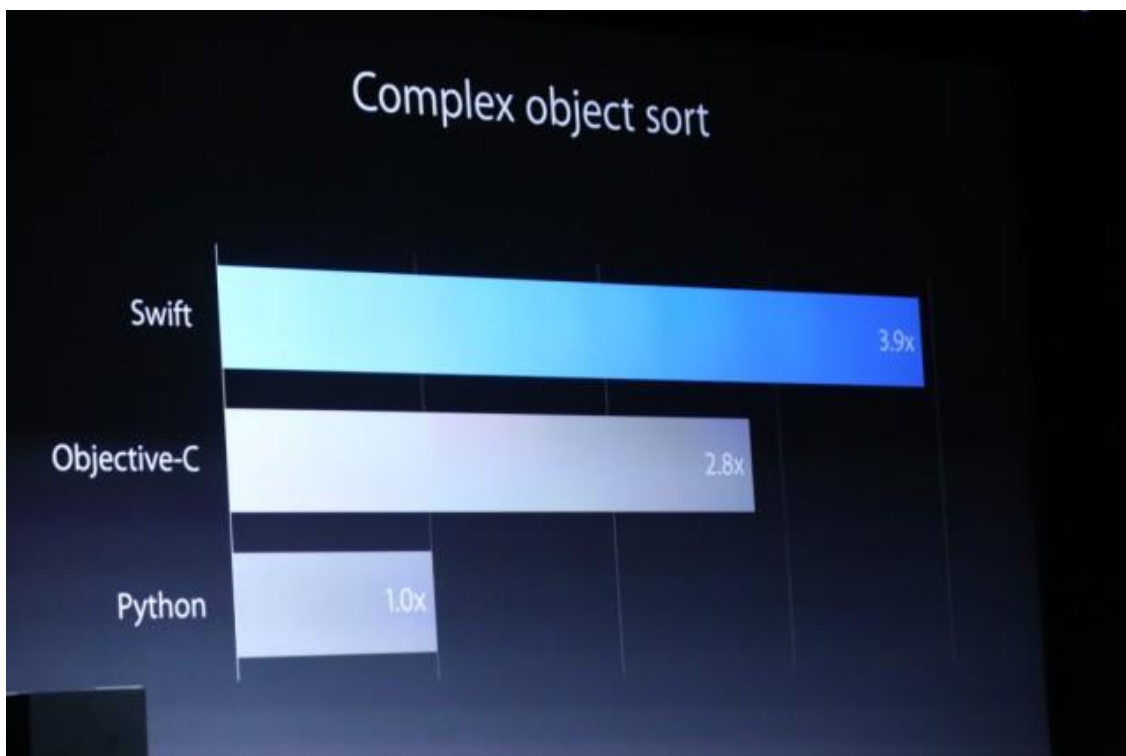
I když je jazyk Swift novější a modernější, stále se používá starší Objective – C. Protože většina aplikací na App Store je psaná v Objective – C, firmy stále vyžadují znalost tohoto jazyka. Swift je hodně mladý a občas musí využívat frameworky a knihovny napsané stále v Objective – C. Objective – C byl často vnímán, kvůli jeho syntaxi zápisu, jako jeden z těžších jazyků pro nováčky. S nástupem Swiftu by měla tahle bariéra opadnout a umožnit nováčkům rychleji si osvojit syntaxi jazyka. Pro porovnání si ukážeme deklaraci prázdného pole [6].

Objective – C: NSMutableArray * array = [[NSMutableArray alloc] init];

Swift: let array = [Int]()

Rozdíl je také v počtu souborů při vytváření tříd. U Objective – C se nám vytvořil hlavičkový a implementační soubor. V hlavičkovém souboru jsme deklarovali proměnné a funkce, které jsme chtěli v aplikaci a v implementačním souboru jsme je poté implementovali. V Swiftu se nám vytvoří jediný soubor, který zastává obě funkce a tím se nám sníží počet souborů, které bychom museli udržovat [6].

Swift je rychlejší v určitých operacích než Objective – C. Je například rychlejší v komplexním srovnávání dat [6].



Obr. 1: Komplexní řazení objektů [6]

2.7 Objektové programování

Objektové programování je založeno na tom, že určité atributy a metody můžeme přiřadit určitému objektu a k těmto atributům a metodám se pak dostáváme přes onen objekt. Kód se pak stává přehlednější a udržitelnější. Třída je vzorem pro objekt a definuje atributy a metody, které bude objekt mít [17].

2.8 App Store

App Store je online obchod sloužící k nakupování a stahování aplikací pro Apple zařízení. Nejdříve byl App Store jen pro zařízení s operačním systémem iOS, později rozšířil také nabídku pro počítače s operačním systémem MacOS X. Pro stahování je nutné mít vytvořené Apple ID. Na App Store je k dispozici více než 500 000 aplikací [7].

Vývojáři, aby mohli dávat svoje aplikace na App Store musí zaplatit každý rok 99\$. Tento poplatek, a přísné podmínky schvalovacího procesu způsobuje, že je na App Store výrazně méně aplikací než na konkurenčním Google Play. Oproti Google Play je

většina aplikací kvalitní a téměř neobsahuje žádné viry ani malware. Z prodeje aplikace dostává vývojář 70% zisku a zbylých 30 % dostává Apple [8].

2.9 XCode

XCode je vývojové prostředí (Integrated development environment) a je dostupný pouze na zařízeních od Applu. Nejnovější verze je 8.2. Slouží k vývoji aplikací pro iPad, iPhone, Apple Watch a Mac. XCode podporuje jazyky Swift, Objective – C, C a C++. S příchodem jazyka Swift je k dispozici hřiště (Playground), kde si může vývojář vyzkoušet syntaxi jazyka Swift [8].

2.10 CocoaPods

CocoaPods je nástroj pro snadné importování knihoven do Swift a Objective – C projektů. Aktuálně je k dispozici přes 29 000 knihoven. Nástroj se ovládá přes příkazový řádek [10].

\$ sudo gem install cocoapods – příkaz pro instalaci CocoaPods

\$ pod init – příkaz pro vytvoření podu, kde specifikujeme, které knihovny chceme do projektu importovat

\$ pod install – příkaz pro instalaci vybraných knihoven

2.11 Knihovny

Seznam knihoven, které budou použity v následujícím projektu.

Alamofire je knihovna, která ulehčuje posílání a přijímání výzev ze serveru a dělá kód přehlednější. Málokterá aplikace nekomunikuje se serverem a díky alamofire je tato problematika o něco příznivější. Alamofire je nejrozšířenější knihovnou pro jazyk Swift. Odezva, kterou dostaneme ze serveru a předá nám ji alamofire je ve formátu JSON [11].

SwiftJSON je knihovna, která ulehčuje zpracování odezvy ze serveru ve formátu JSON. Díky této knihovně je kód o něco přehlednější a kratší. Zároveň se stává mnohem stabilnější [12].

SWRevealViewController je knihovna, která je napsaná v Objective – C, ale díky komptabilitě není problém ji importovat do Swift projektu. Jedná se o knihovnu, která schová obrazovku do pozadí, kterou můžu dát kdykoliv do popředí. V mé práci se bude jednat konkrétně o menu, které bude na každé obrazovce v pozadí a po zmáčknutí tlačítka vyjede zprava a dostane se do popředí. Jedná se o efekt, který je čím dál rozšířenější a můžeme ho najít například v aplikaci Facebook [13].

BRYXBanner je knihovna, která umožňuje zobrazovat notifikace ze serveru, zatímco je aplikace v popředí [14].

2.12 SWOT analýza

K analýze současného stavu použijí SWOT analýzu. SWOT analýza je univerzální nástroj, který zhodnocuje vnitřní a vnější faktory, které mohou ovlivnit úspěch daného záměru. Analýza se dělí na čtyři složky, silné stránky, slabé stránky, příležitosti a hrozby [15].



Obr. 2: SWOT analýza [16]

3 Analýza současného stavu

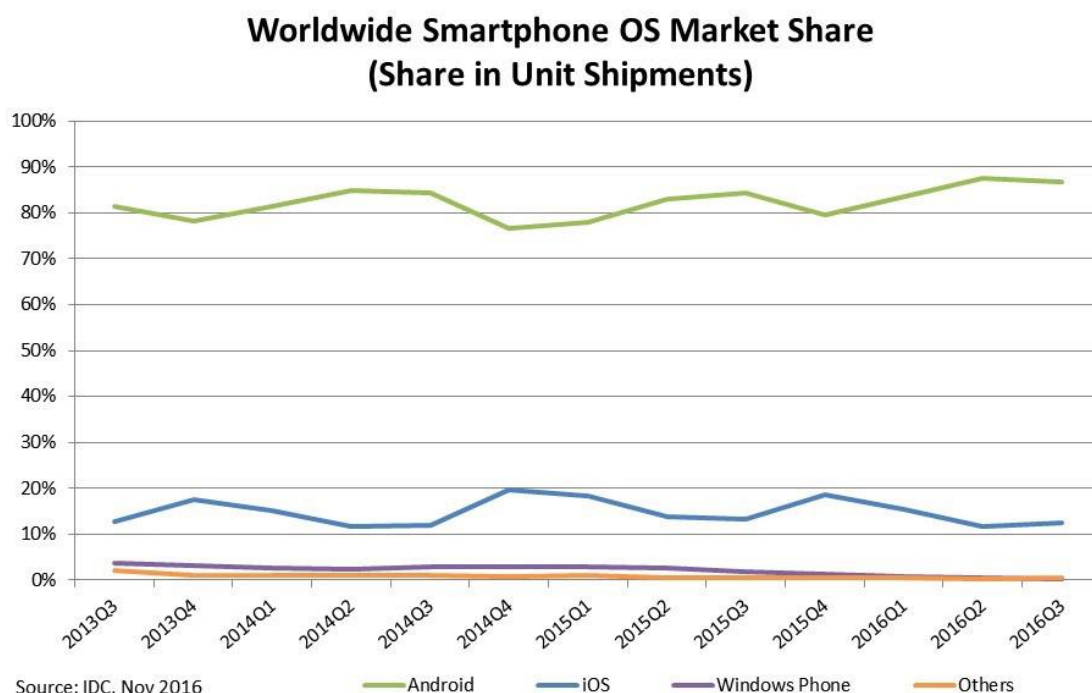
Analýza současného stavu je důležitá pro plánování a vytváření strategií. Správná analýza nám dokáže pomoci k vytvoření produktu, který bude konkurenceschopný na trhu. V této části práce zanalyzuji trhy pro platformy iOS, Android a Windows Phone. Dále rozeberu možnosti monetizace, jejich výhody a nevýhody. Poté vytvořím SWOT analýzu, analyzuji stránku gaynet.ch a nakonec shrnu výsledky analýzy.

3.1 Podíl mobilních platforem na trhu

V současnosti má nejvíce dominantní postavení na trhu Android s podílem 86,8 %. To je o 7,2 % více než minulý rok. Je to způsobeno především klesající popularity ostatních platforem. K tomuto vysokému podílu na trhu také nejvíce přispívá fakt, že je Android open source, a proto většina společností, které vyrábějí telefony ho využívají jako operační systém. To nabízí zákazníkovi rozmanitý výběr telefonů, od levných a méně výkonných až po vysoce výkonné. Nejrozšířenějším obchodem pro Android je Google Play, na kterém uživatelé stáhli dohromady 65 miliard aplikací za dobu její existence [18].

Druhou platformou, která má významnou pozici na trhu je iOS s podílem 12,5 %. Platforma iOS je velice uzavřená, a dostupná pouze na produktech od společnosti Apple.

Platforma Windows Phone v poslední době upadá a v současnosti reprezentuje pouze 0,3% podílu na trhu. Ještě před rokem to bylo 1,2 %. Kvůli takto malému podílu se dále nebudu zabývat touto platformou a pro tuto práci budeme analyzovat pouze obchody Google Play a App store.



Obr. 3: Rozdělení světového trhu podle OS [19]

3.2 Google Play

Jak už bylo zmíněno Google Play je největším obchodem pro Android aplikace. Má mnoho kategorií, mezi kterými může uživatel vybírat, avšak nejstahovanější kategorií jsou hry, které v roce 2015 tvořily 21 % všech stáhnutých aplikací a 90 % příjmů [20], [21]. Vytvoření mobilní hry a poté umístění na Google Play je tedy dobrou příležitostí, jak získat určitý zisk a prorazit na scénu mobilních her.

K umístění aplikace na Google Play je potřeba založit Google Play publisher account, zadat údaje vývojáře a poté zaplatit poplatek 25\$ [21].

3.3 App Store

Stejně jako u Google Play jsou na App Store hry nejrozšířenější kategorií. K roku 2016, hry tvoří skoro 25 % stahovaných aplikací a 75 % příjmů. [22], [24]. I když je počet zařízení s Androidem mnohonásobně vyšší, App Store má o 90 % vyšší tržby [24]. Může za to několik faktorů. Jedním z nich je Čína, která je první s útratou 1,7 miliard dolarů [25]. Dalším z nich je, že většina aplikací na Google Play vydělává pomocí reklam, které nejsou zahrnuty.



Obr. 4: Top 3 Nejvíce utrácující země na App Store [24]

K umístění aplikace na App Store je potřeba založit vývojářský účet a zaplatit roční poplatek 99\$. Aplikace nahraná na App Store se musí nejdříve nechat schválit a až poté je zveřejněna. Apple je velice striktní ohledně nových aplikací. K tomu, aby aplikace mohla být zveřejněna, musí být kvalitní, originální, funkční a bez urážlivých motivů [23].

3.4 Monetizační modely

Možností, jak získat příjem z aplikace je několik a často vybrat tu správnou je velice klíčové. U některých produktů se stává, že je monetizace špatně nastavená a musí se přejít z jednoho modelu na jiný.

3.4.1 Zobrazení reklam

Na Google Play se jedná o nejčastější model. Uživatel může využívat veškerý obsah aplikace zdarma, ale zobrazuje se mu reklama. Forma reklamy se často liší. Může jít o reklamu, která se zobrazuje tak, aby uživatele nerušila při užívání aplikace nebo o vyskakovací reklamu, kterou uživatel musí zavřít, až po videa, která uživatel může odkliknout po přehrání. Interval zobrazení reklam je stejně důležitý jako kontent a příliš časté zobrazení reklam může odradit většinu uživatelů. Zisk je generován po zobrazení

reklamy, je tedy důležité, aby uživatel aplikaci používal co nejvíce. Model může být spojen s freemium modelem, kdy zaplatíte určitou částku, aby se vám reklamy přestaly zobrazovat [27].

Výhody

Aplikace je pro uživatele zdarma a můžou si tedy aplikaci sami vyzkoušet.

Nevýhody

Zisk je závislý na častém užívání aplikace, tedy po stáhnutí nemáte jistotu, že vaše aplikace bude používána a bude generovat zisk.

3.4.2 Placená aplikace

Uživatel jednorázově zaplatí za aplikaci a poté má k dispozici veškerý obsah. Zisk tedy závisí na počtu stáhnutí, nikoliv na užívání [27].

Výhody

Uživatele neobtěžují reklamy a po zakoupení nemusí dále platit. Vývojář získá peníze ihned po zakoupení.

Nevýhody

Většina uživatelů nechce zaplatit za produkt, který si nemůže vyzkoušet předem, musí se tedy spoléhat na recenze od ostatních zdrojů. Nevhodné pro typ aplikací, které musí vývojář aktualizovat, protože zisk je jednorázový.

3.4.3 Nákupy v aplikaci

Jedná se o model, který se vyskytuje především v herní sekci. Uživatel si může během hraní koupit virtuální věci, které mu pomůžou ve hře. Uživatel může například získat

výhodu nad protihráčem, dokoupit životy, aby mohl pokračovat v hraní, upravit kosmeticky svoje postavy a podobně. Mnoho z těchto modelů staví na tom, že většina uživatelů hraje jejich hry zadarmo a pak je pár hráčů, velryb (angl. Whales), kteří pumpují do těchto her velké množství peněz. Tito hráči získají velkou výhodu nad ostatními hráči a finančně živí další vývoj aplikace [27].

Výhody

Pro uživatele je aplikace dostupná zdarma a většinou bez reklam. Uživatel si sám může vybrat, zda ho aplikace baví natolik, že by do ní investoval.

Nevýhody

Některé uživatele může odradit, že ostatní mají nad nimi výhodu jen díky tomu, že zaplatili víc.

3.4.4 Freemium

Model freemium nabízí omezený obsah, který může být rozšířen. Uživatel si zdarma stáhne aplikaci a poté zda je spokojen, může zaplatit za prémiový rozšířený obsah [27].

Výhody

Uživatel může aplikaci vyzkoušet zdarma a v případě libosti dokoupit prémiový obsah.

Nevýhody

Občas je složité najít rovnováhu, co zpřístupnit zdarma a co za peníze. Aplikace by měla být atraktivní i když je zdarma, ale zároveň mít obsah za který se vyplatí zaplatit.

3.4.5 Předplatné

Předplatné je podobné jako freemium. Určitý obsah je zdarma, ale za celkový obsah se musí zaplatit. Rozdíl je v tom, že předplatné umožní obsah jen na určitou dobu, nejčastěji jeden měsíc a poté se musí zaplatit znovu. Tento model se často používá u aplikací, které nabízí široký obsah a jsou finančně náročné na provoz, například streamovací služby jako Spotify, Apple Music nebo Netflix [27].

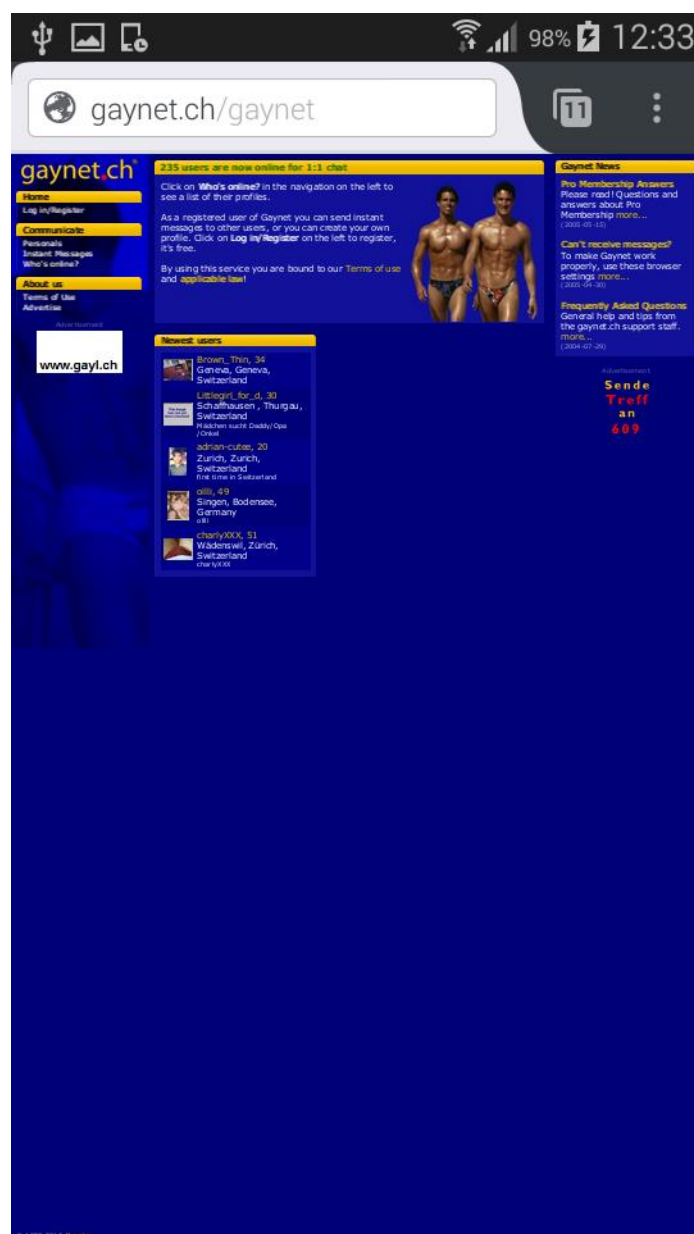
Výhody

Uživatel má po zaplacení přístup k veškerému obsahu. Zejména u streamovacích služeb má přístup k velké databázi skladeb nebo seriálů.

Nevýhody

Obsah je zpřístupněn jen na určitou dobu. Uživatelé s menšími příjmy budou volit alternativní služby.

3.5 Analýza stávající webové stránky



Obr. 5: Gaynet [26]

Pokud se rozhodneme zobrazit si službu Gaynet na mobilních zařízeních, zjistíme, že webové stránky nejsou responzivní, tedy že se zobrazují stejně, jako by se zobrazovaly na monitoru. Veškerý text není čitelný, pokud ho ručně nepřiblížíme. Po přiblížení je těžké se na stránce orientovat, protože uživatel musí přejíždět jak vertikálně, tak i horizontálně. Tlačítka se špatně klikají a většinou se stane, že uživatel klikne sousední tlačítko a musí se vrátit.

Webová stránka vznikla v roce 1999 a od té doby se nezměnila. Kvůli tomu vypadá velmi zastarale a na dnešní poměry hodně amatérsky. I přes zastaralý design, které by mnohé mohl odradit zde můžeme vidět 235 online uživatelů.

3.6 SWOT Analýza

Silné stránky <ul style="list-style-type: none"> - Aplikace je nativní - Aplikace je zdarma - Webovou službu už využívá mnoho uživatelů 	Slabé stránky <ul style="list-style-type: none"> - Aplikace je zaměřena pouze na švýcarský trh - Pouze pro iOS
Příležitosti <ul style="list-style-type: none"> - Popluarita aplikace - Reference do portfolia 	Hrozby <ul style="list-style-type: none"> - Aplikace se neuchytí - Aplikace nebude povolena na App Store

Tab. 1: SWOT analýza

Silné stránky

Aplikace je vyvíjena nativně, to znamená, že je vyvíjena pouze pro iOS. S tím souvisí lepší běh aplikace, bude více stabilní a padat bude v ojedinělých případech. Zásahy do kódu budou méně rizikové a problémové. Kód bude více optimalizovaný. Aplikace bude běžet rychleji a spotřebovávat méně baterie.

Aplikace bude zdarma, a tedy dostupná všem. Zisk bude generován ze zobrazení reklam a zakoupením prémiového účtu, který zpřístupní některé funkce, například prohlížení soukromých fotografií.

Díky tomu, že je webová služba zavedená již několik let, bude lehčí získat uživatele, kteří budou tuto aplikaci využívat. Mnoho aplikací má problémy, protože nemá základnu uživatelů, kteří by ji využívali, a tak některé aplikace, které by mohly mít úspěch skončí.

Slabé stránky

Aplikace je zaměřena pouze na švýcarské uživatele a uživatele v jejím okolí. Tím je omezeno publikum, ke kterému se aplikace dostane. Pokud by se aplikace soustředovala na uživatele ostatních zemí, mohla by přitáhnout větší publikum a tím zvýšit zisky.

Protože je tahle aplikace vyvíjena nativně pouze pro iOS, a trh pro Android je příliš velký, aby se dal ignorovat, bude nutné vyvinout druhou aplikaci pouze pro Android. Existují nástroje, které dokáží vyvinout jednu aplikaci, která funguje pro iOS i Android, ale tato aplikace není tolik stabilní a je více problémová.

Příležitosti

U této aplikace se počítá s určitou základnou uživatelů, kteří budou tuto aplikaci využívat. Je zde ale příležitost, že by se aplikace dobře uchytila a mohla přilákat mnoho dalších aktivních uživatelů.

Jako student, který se začíná pohybovat na poli mobilních aplikací, bude tato aplikace skvělou referencí v portfoliu.

Hrozby

Je zde hrozba, že mnozí uživatelé jsou již zvyklí na webovou službu a nebudou si chtít aplikaci ani vyzkoušet. Sice se očekává, že aplikace bude hojně využívána, ale tato hrozba zde je.

Aplikace, které se schvalují na App Store prochází přísným procesem kontroly. Aplikace musí být originální, bez chyb a kvalitní. Oproti Google Play, kde schvalování není tak přísné zde hrozí, že aplikace nemusí být umístěna na App Store.

3.7 Shrnutí

Webovou službu využívá značný počet uživatelů, který by se mohl i nadále rozrůst. Proto bude vytvořena aplikace pro iOS, která zaručí pohodlný přístup i ze svého

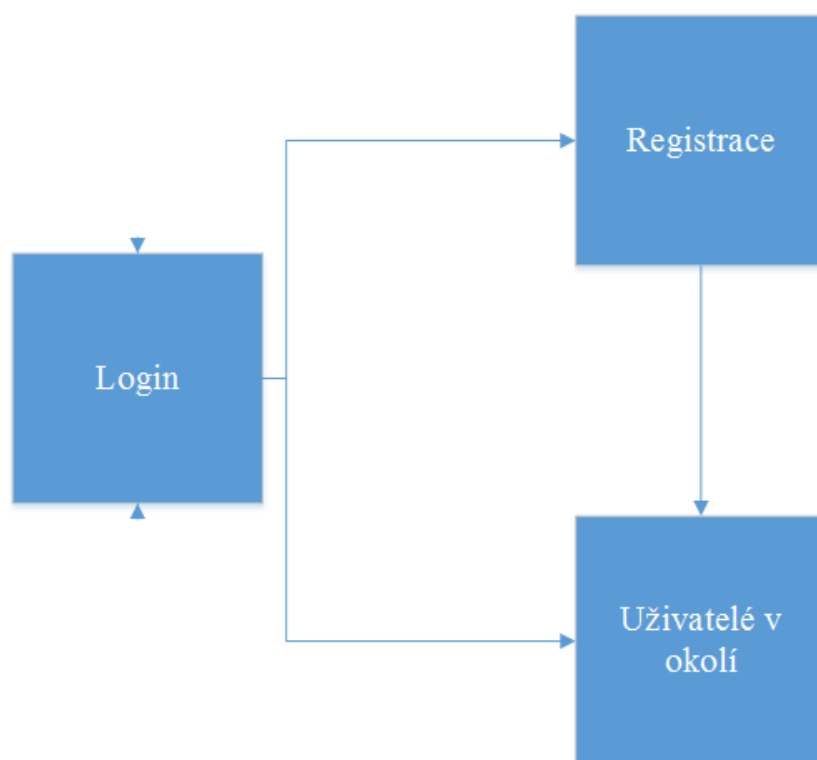
mobilního zařízení a bude se snažit získat nové uživatele. Jako monetizační model bude možnost koupě prémiového účtu, který umožní zobrazovat nepřístupné fotografie a mnoho dalšího. Aplikace bude zdarma, a tak si ji může stáhnout a vyzkoušet každý.

4 Vlastní návrh řešení, přínos práce

Při zakládání projektu v programu Xcode je důležité určit, pro které platformy bude aplikace vyvíjena, v kterém jazyce bude napsána a zda bude využívat ukládání dat pomocí Core Data. V našem případě bude aplikace vyvíjena pouze pro iPhone v jazyce Swift a nebude využívat Core Data, jelikož bude vše ukládáno na straně serveru a jednotlivé informace se budou získávat pomocí výzev. Aplikace bude požadovat, aby uživatelé měli iOS verzi 8.0 a vyšší. Uživatelé s nižší verzí nebudou moci stáhnout tuto aplikaci a budou nuceni upgradovat na požadovanou verzi. Je to z toho důvodu, že většina knihoven je napsaná pro verzi 8.0. Operační systém iOS 8.0 je podporován od iPhone 4S a výš a v iPodech Touch 5. a 6. generace. Starší iPhony nebudou mít možnost stáhnout tuto aplikaci. Aplikace bude mít možnost zjistit současnou lokaci uživatele, přístup k fotografiím a videím, ke kameře a posílání notifikací. Lokace uživatele je důležitá pro zjištění, zda se nachází ostatní uživatelé v okolí. Přístup k fotografiím, videím a ke kameře je pro umožnění nahrání fotek a videí na svůj profil. Notifikace upozorňují na veškeré události, které by daného uživatele mohly zajímat.

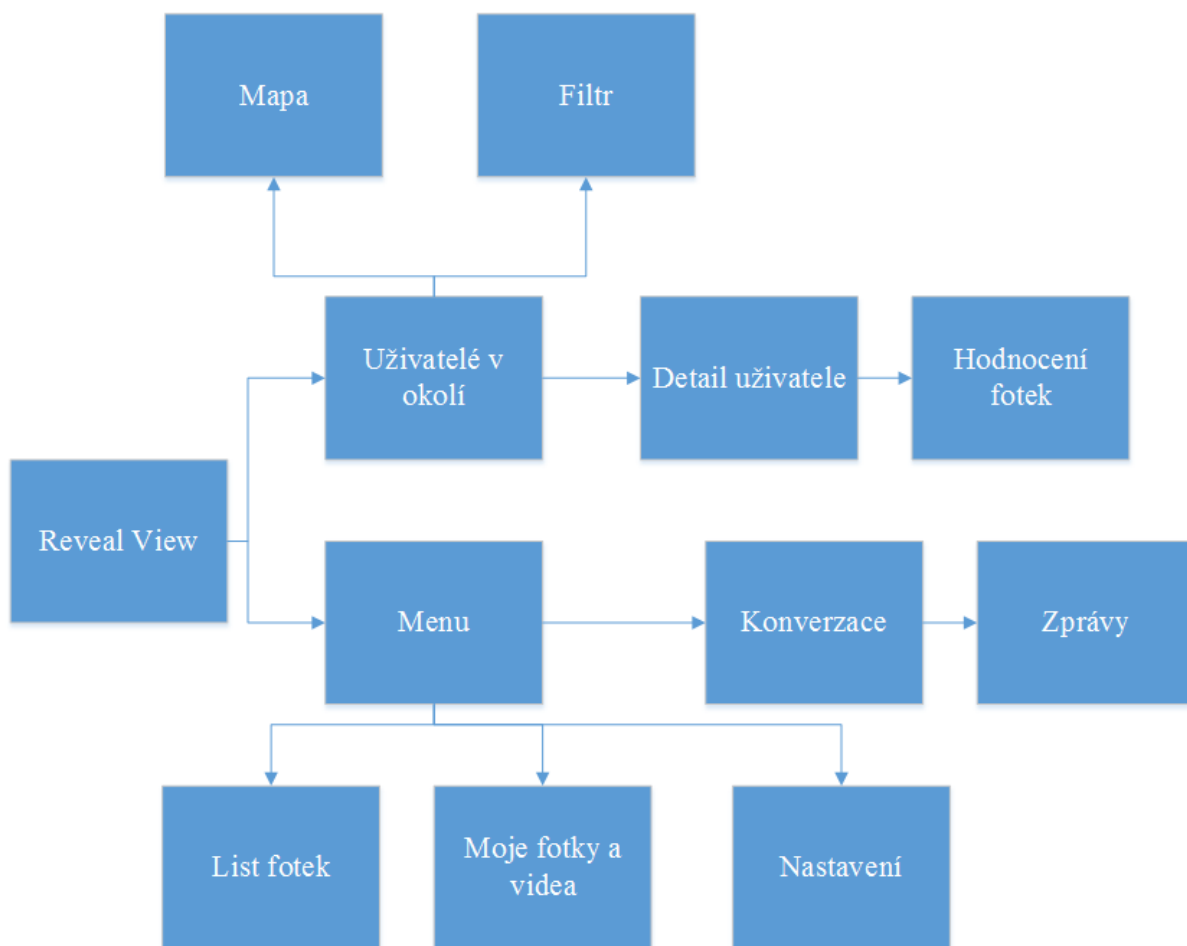
Aplikace se stále vyvíjí a testuje, a proto je napojena pouze na testovací API, aby nezasahovala do dat skutečných uživatelů. Veškeré data, která budou zobrazena jsou pouze demonstrativní.

Než začneme psát kód, je důležité si uvědomit, které obrazovky bude aplikace potřebovat a jaká bude jejich logika. K lepšímu porozumění nám bude sloužit diagram, který zobrazí veškeré obrazovky a jejich logiku. Hned úvodní obrazovka bude přihlašovací, kde uživatel zadá svoje přihlašovací údaje a poté, zda je vše v pořádku je přenesen na obrazovku uživatelé v okolí. V případě že uživatel nemá přihlašovací údaje, je mu nabídnuto se registrovat a po vyplnění údajů je přenesen na obrazovku uživatelé v okolí, stejně jako u přihlašování.



Obr. 6: Diagram přihlašování

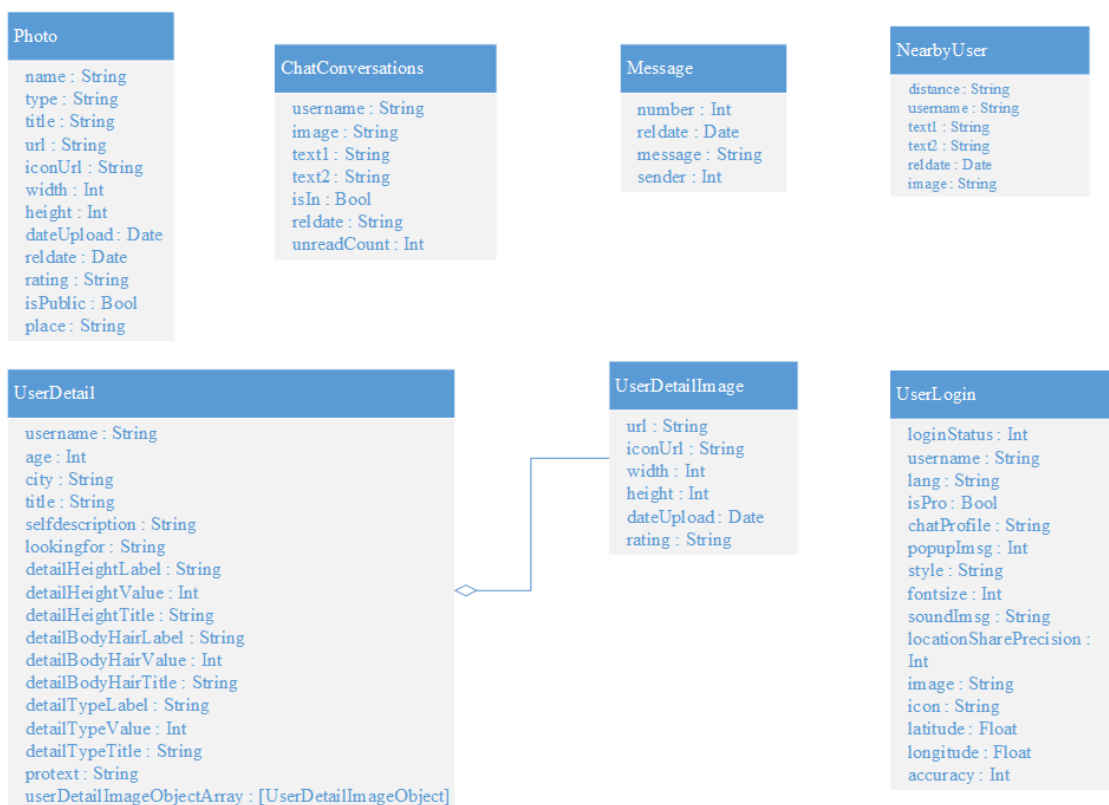
Ještě před tím, než se zobrazí uživatelé v okolí je zavolána obrazovka Reveal View, která dá do pozadí menu a do popředí obrazovku, která se má načíst, tedy uživatelé v okolí. Na obrazovce uživatelé v okolí je několik možností, kudy se vydat. Uživatel si může zobrazit detailní profil jiného uživatele, který je v okolí. Může si zobrazit mapu, na které uvidí, jak daleko jsou ostatní uživatelé. Může si otevřít filtr, a podle určitých parametrů vyfiltrovat ostatní uživatele. Poslední možností je otevřít menu, které zase otevře několik možností. Z obrazovky detail uživatele se dále můžeme dostat na hodnocení fotek. Z menu se dále můžeme dostat na seznam konverzací s jednotlivými uživateli a poté na konkrétní konverzaci. Poslední obrazovky, na které se z menu můžeme dostat jsou list fotek, moje fotky a videa a nastavení. Poslední položkou v menu je odhlášení, které nás opět přesměruje na obrazovku přihlášení.



Obr. 7: Hlavní diagram

4.1 Návrh tříd

Při každém volání na server dostaneme výsledek dat v JSONu. Tyto data, abychom s nimi mohli dále pracovat, potřebujeme někde uchovávat. K tomu nám poslouží instance tříd, které budeme vytvářet při každém volání. Každá třída je navržena tak, aby uchovala veškeré informace, které ze serveru dostane. Třída `UserDetail` obsahuje pole instancí třídy `UserDetailImage`, proto je mezi nimi vztah agregace.



Obr. 8: Návrh tříd

4.2 Přihlašovací obrazovka

Jednoduchá přihlašovací obrazovka, která má dvě textové pole. Jedno pole na zadání emailu a druhé na zadání hesla. Obrazovka také obsahuje tlačítko, které vezme údaje v textových polích a odešle požadavek na server. Server vrátí výsledek a pokud je status kód 200, je uživatel vpuštěn dál. Veškeré informace, které vrátí server se uloží do objektu UserLogin. Jde především o jméno, profilovou fotku, zda má uživatel prémiový účet a polohu uživatele, pokud je známa. Script přihlašovací obrazovky kontroluje, zda jsou obě pole vyplněná, a zda mají hodnoty v poli požadovanou délku a tvar. Pole pro email musí obsahovat zavináč a pole pro heslo musí mít minimálně 6 znaků. Dále se zde nastavuje navigační bar, který bude dále pro všechny obrazovky stejný. Jedná se o tmavě modrou lištu nahoře.

••••• O2 - CZ 12:39 100 %

Login

Email

Password

LOGIN

REGISTER

Obr. 9: Přihlašování

4.3 Registrační obrazovka

Registrační obrazovka obsahuje pět textových polí, jeden switch a jedno tlačítko. Uživatel zadá svoje jméno a heslo, datum narození, email, a ještě jednou email pro potvrzení. Pro zadání datumu narození se místo klávesnice vysune date picker, který umožňuje vybrat určité datum. Script opět kontroluje, zda není některé pole prázdné, email a heslo jsou ve správném tvaru a uživatel je starší nad 18 let, jinak vyhodí chybovou zprávu. Dále je zde odkaz na podmínky se kterými musí uživatel souhlasit. Poté co uživatel vyplní všechna pole je poslán požadavek na server, který vrátí výsledek. Pokud je vše v pořádku a status kód je 200, pošle so nový požadavek na server s přihlášením, kde se vyplní jméno a heslo zadané při registraci a poté je uživatel vpuštěn dále. Stejně jako u přihlašování jsou veškeré informace uloženy do objektu UserLogin.

Registration

Username

Password

Birthday

Email

Email again

Terms and conditions

REGISTER

Obr. 10: Registrace

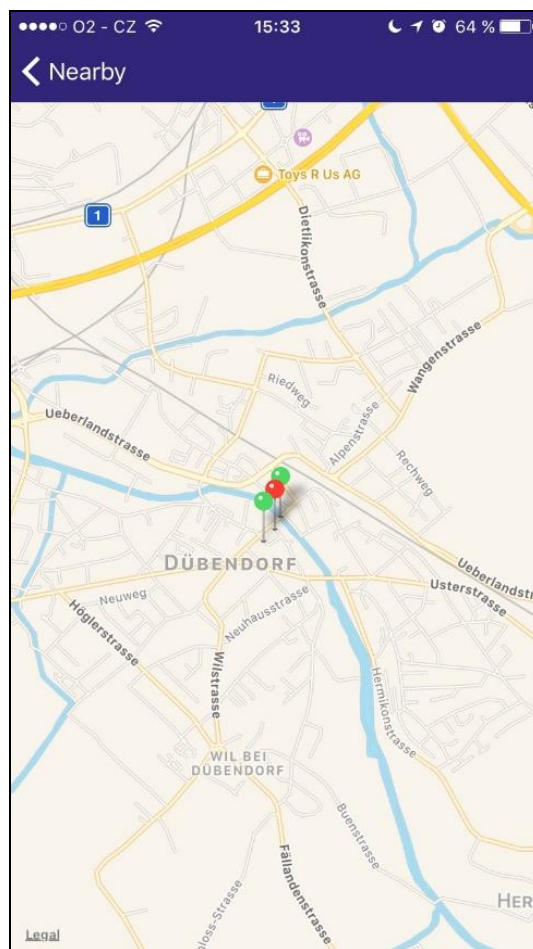
4.4 Uživatelé v okolí

Pokud je známa uživatelská poloha, pošle se požadavek na server. Požadavek obsahuje zeměpisnou šířku a délku a také přesnost polohy. Server poté vrátí seznam uživatelů, kteří jsou v okolí a informace o nich. Každý uživatel v okolí je uložen do objektu `NearbyUser` a tyto objekty jsou poté uloženy do pole pro snadnější práci s objekty. Poté co jsou uživatelé uloženi v poli jsou zobrazeni v kolekci. Každá buňka kolekce obsahuje profilovou fotku uživatele, jméno a jak je daleko. Po kliknutí na kteroukoliv buňku se otevře obrazovka s detailními informacemi o uživateli. Fotky uživatelů se stáhnou pouze jednou a poté jsou uloženy do cache paměti. Pokud je požadavek stáhnout stejnou fotku, vytáhne se tahle fotka z cache paměti a ušetří se přenosová data. V navigačním baru jsou tlačítka, které převedou uživatele na mapu nebo filtr.



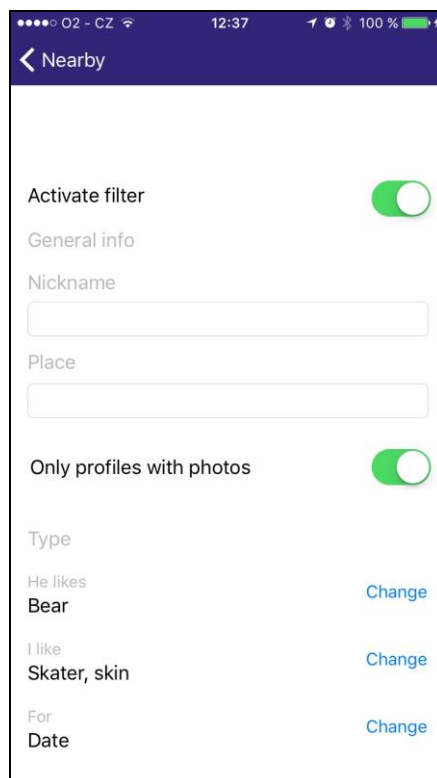
Obr. 11: Uživatelé v okolí

Obrazovka mapa ukazuje na mapě ostatní uživatele. Pro zobrazení těchto uživatelů je nutná jejich zeměpisná šířka a délka. Tyto informace jsou uloženy v poli s `NearbyUser` objekty. Každého uživatele poté vykreslíme na mapě včetně přihlášeného uživatele a přiřadíme mu příslušný špendlík. Přihlášený uživatel má červený špendlík, ostatní uživatelé mají zelený. Po kliknutí na špendlík se zobrazí okénko s jménem a věkem uživatele a jak je daleko.



Obr. 12: Mapa uživatelů v okolí

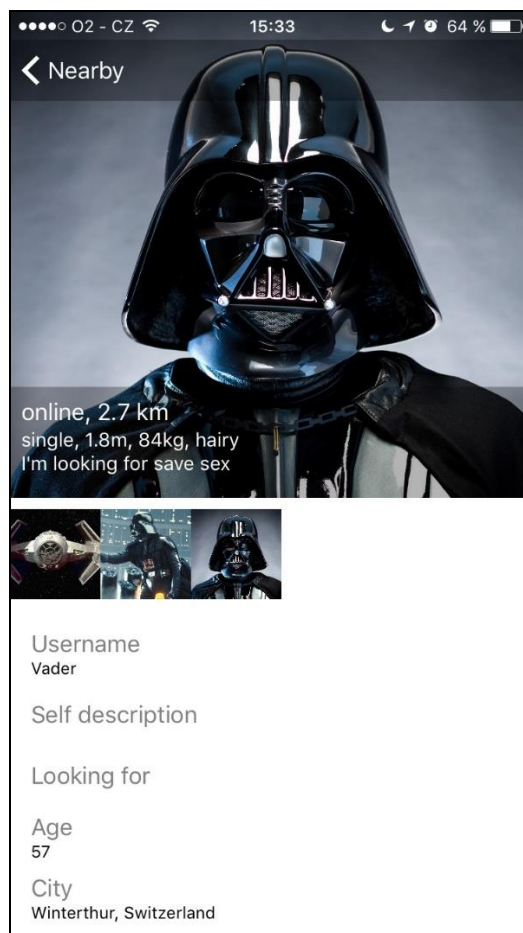
Obrazovka filtr obsahuje pole, které může uživatel vyplnit a poté se odešle požadavek na server, který vrátí filtrované uživatele v okolí. Filtr obsahuje dvě textové pole, dva switchy a několik položek, podle kterých si uživatel může upravit vyhledávání. Všechny tyto změny se pošlou jako parametry na server.



Obr. 13: Filter

4.5 Detail uživatele

Pro zobrazení detailů konkrétního uživatele je poslán požadavek na server s jeho jménem. Po vrácení odpovědi ze serveru jsou data uložena do objektu `UserDetail`. Tyto data jsou následně zobrazena na obrazovce. Vrchní část obrazovky se skládá z profilové fotky, která je v pozadí a na popředí jsou důležité informace o uživateli. Tyto informace mají černé průhledné pozadí pro případ, že má fotka bílé pozadí. S bílým textem by to bylo nečitelné. Pod profilovou fotkou se nachází kolekce fotek. Kolekce fotek se dá posouvat pouze horizontálně a zobrazuje všechny veřejné fotky uživatele. Po kliknutí na některou z fotek je uživatel přesunut na hodnocení fotky. Pod kolekcí fotek je list s ostatními informacemi o uživateli. Tato obrazovka má jako jedna z mála průhledný navigační bar pro lepší estetičnost obrazovky.



Obr. 14: Profil uživatele

4.6 Hodnocení fotek

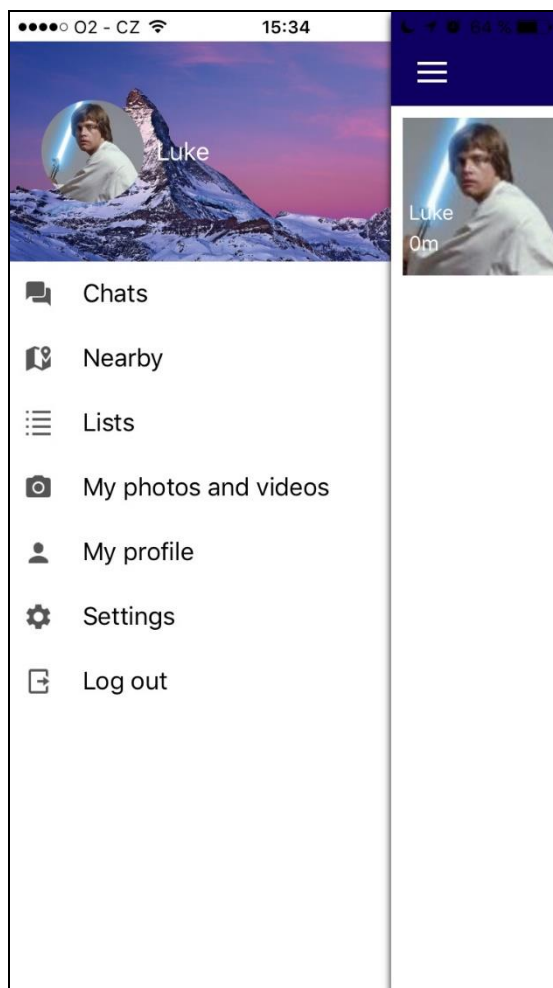
Po tom, co uživatel klikne na jednu z fotek z kolekce se otevře obrazovka hodnocení fotek. Uživatel může mezi fotkami přejíždět a hodnotit. Aby přejíždění bylo co nejvíce responzivní, vypočítá se šířka jednotlivých fotek a poté se mimo obrazovku naskládají vedle sebe. Na pozadí obrazovky je pak posunovací obrazovka, která se stará o to, aby se zobrazila další fotka, která bude na středu a nebudou na obrazovku zasahovat ostatní fotky. Stejně jako u předešlé obrazovky je také navigační bar průhledný. Veškeré fotky, které se jednou stáhnou budou uloženy v cache paměti, takže je nebude potřeba stahovat znovu.



Obr. 15: Hodnocení fotek

4.7 Menu

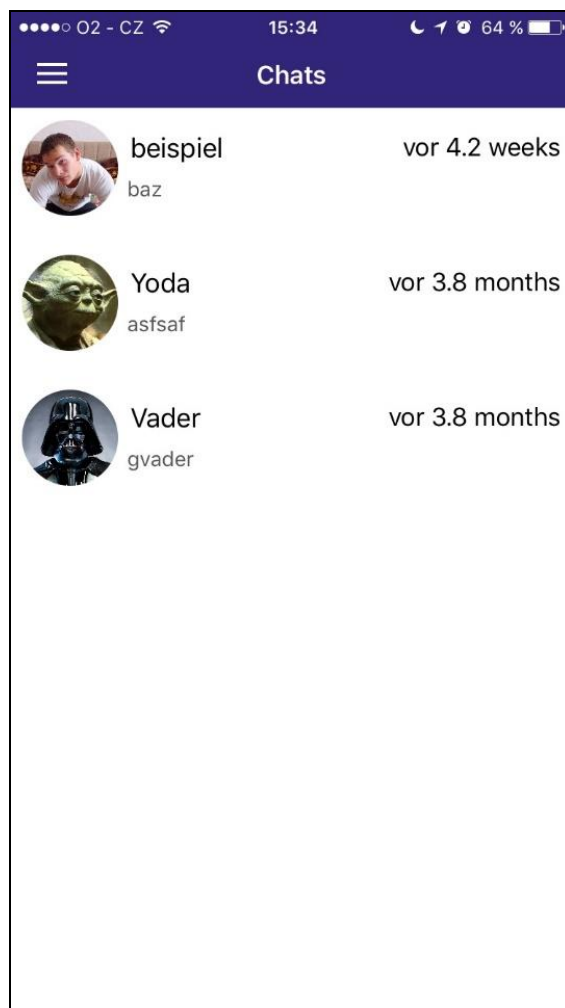
Menu obsahuje list položek, které odkazují na určité obrazovky. Menu je speciální obrazovka, která je vždy v pozadí ostatních obrazovek a při zmáčknutí menu tlačítka se dá na popředí a vyjede zleva. O tuto funkcionalitu se stará knihovna `SWRevealViewController`.



Obr. 16: Menu

4.8 Konverzace

Obrazovka s konverzacemi obsahuje list, který zobrazuje jednotlivé konverzace s uživateli. Při načtení obrazovky se pošle požadavek na server, který vrátí seznam konverzací. Po kliknutí na určitou buňku listu se zobrazí zprávy, které si dotýčný uživatel posílá s jiným uživatelem. Informace, které dostane aplikace ze serveru se uloží do objektu ChatConversations a tyto objekty se uloží do pole. Každá buňka obsahuje jméno, poslední zprávu, profilovou fotku a kdy byla naposledy přijatá zpráva. Po přejetí určité položky zprava do leva bude nabídnuta uživateli možnost tuto konverzaci smazat.

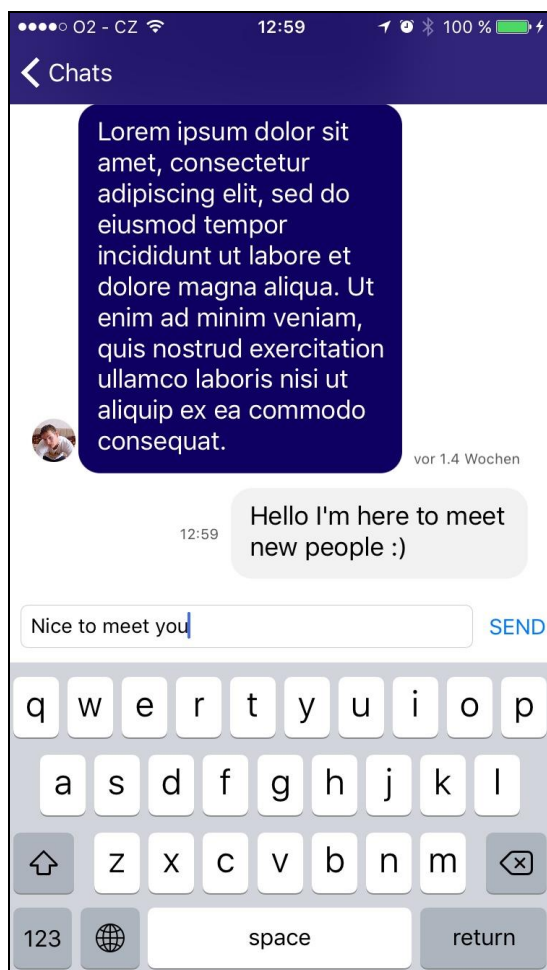


Obr. 17: Konverzace

4.9 Zprávy

Pro zobrazení zpráv se pošle požadavek na server, který obsahuje jméno, počet zpráv a pozici daných zpráv. Je to kvůli tomu, aby se pokaždé nestahovaly všechny, ale pouze část. Sníží se tím zátěž na internetová data a obrazovka nebude muset najednou načítat tolik dat. Celá obrazovka je kolekce, a každá její jedna buňka je jedna zpráva. Protože se délka všech zpráv liší, musí se upravit i chování buněk. Každá buňka si vypočítává délku zprávy a podle toho si upravuje vlastní velikost. Dále je potřeba rozlišovat zprávy příchozí a odchozí. O to se stará sama kolekce, která umísťuje buňky nalevo nebo napravo. Ke každé buňce je také přidělena profilová fotka a čas kdy byla zpráva odeslána. Pod kolekcí se nachází textové pole, kde může uživatel napsat zprávu a odeslat ji. Problém nastává, kdy uživatel chce napsat zprávu a vyjede ze spodu klávesnice, která zakryje textové pole. Proto je ve skriptu funkce

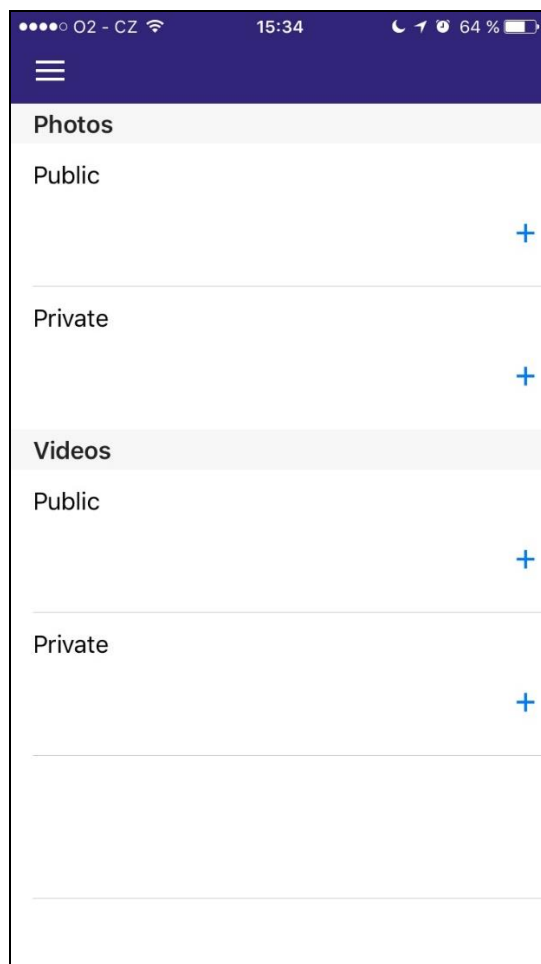
handleKeyboardNotification, která při kliknutí na textové pole spočítá velikost klávesnice a posune textové pole výš, aby mohl uživatel vidět co píše.



Obr. 18: Zprávy

4.10 Moje fotky a videa

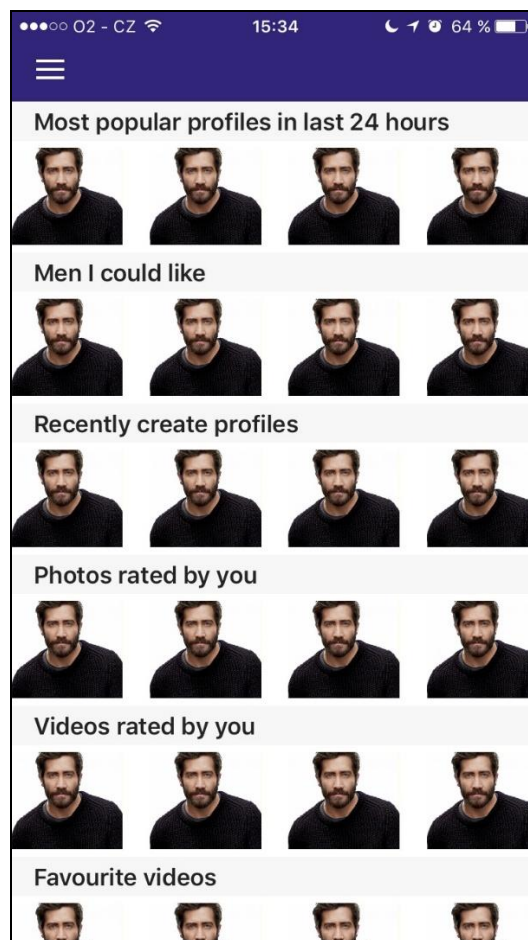
Obrazovka moje fotky a videa obsahuje list a každý list obsahuje kolekci fotek nebo videí. List má určený dvě sekce a každá sekce má dvě buňky. Buňky jsou rozděleny na soukromé a veřejné. Každá buňka má tlačítko pro přidávání nového obsahu. Po kliknutí tlačítka se otevře dialog a uživatel si může vybrat, zda chce pořídit novou fotku nebo video nebo vybrat existující z galerie. Po dokončení výběru se odešle požadavek na server pro uložení média. Po kliknutí na určitou fotku se uživatel přesune na obrazovku hodnocení fotek. Po kliknutí na video se video přehraje. Pro přehrávání videí slouží knihovna od Applu AVKit.



Obr. 19: Moje fotky a videa

4.11 Listy fotek

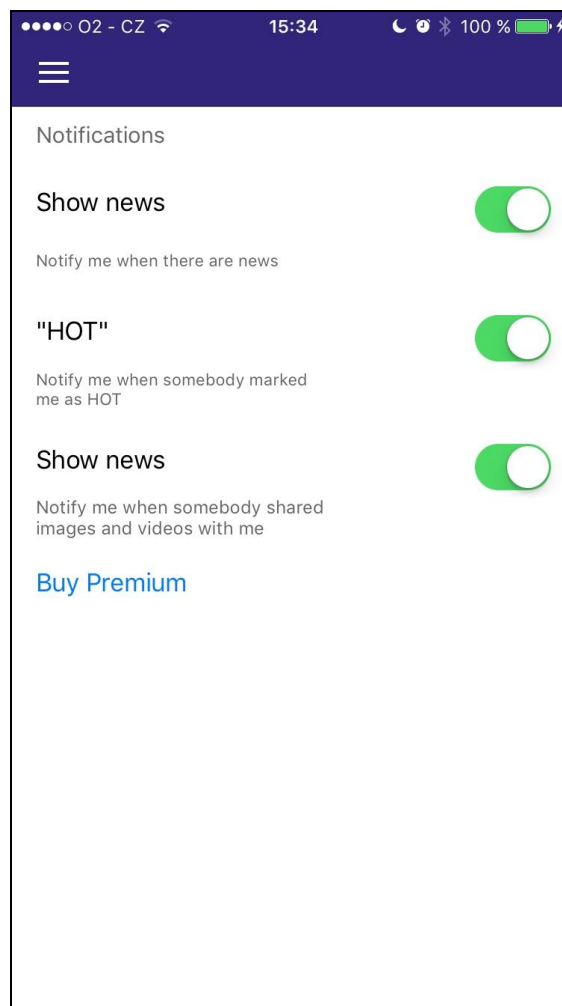
Obrazovka listy fotek stejně jako moje fotky a videa obsahuje list a každá buňka listu obsahuje kolekci fotek. Pro zobrazení se pošle požadavek na server, který vrátí kategorie a fotky uživatelů. Ty se poté zobrazí v určitých kategoriích. Po kliknutí na kteroukoliv fotku je uživatel přenesen na detail uživatele.



Obr. 20: Listy

4.12 Nastavení

Nastavení obsahuje pouze tři switche pro notifikace, které si uživatel může nastavit a tlačítko pro koupení prémium účtu. Registrování notifikací se provádí v třídě AppDelegate, kdy je na server poslán token zařízení a podle toho pak server posílá notifikace. Notifikace se zobrazují pouze, pokud je aplikace v pozadí, a proto musíme implementovat knihovnu BRYXBanner, která nám zobrazí notifikaci, i když bude aplikace v popředí. Po kliknutí na tlačítko Buy Premium je uživatel přenesen na platební bránu, kde vyplní požadované údaje. Po odeslání platby je odeslán požadavek na server, kde se nastaví uživatelův účet na prémiový. Informace o tom, zda má uživatel prémiový účet se dostane z požadavku při přihlášení.



Obr. 21: Nastavení

4.13 Ekonomické zhodnocení

Ekonomické zhodnocení se bude týkat pouze zhotovení ios aplikace. API, která se stará o veškeré požadavky na serveru dodává jiná firma, kterou si klient vybral sám. Celkové provedení aplikace bylo odhadnuto na 100 hodin. Hodinová sazba juniorního programátora byla 200 Kč/h.

Přihlášení	3 hodiny
Registrace	4 hodiny
Uživatelé v okolí	6 hodin
Filtr	6 hodin
Notifikace	4 hodiny
Listy	8 hodin
Menu	3 hodiny
Můj profil	10 hodin
Detail uživatele	12 hodin
Chat konverzace	5 hodin
Chat zprávy	16 hodin
Moje fotky a videa	8 hodin
Nastavení	4 hodin
Napojení na API	11 hodin
Celkově hodin	100 hodin
Hodinová sazba	200 Kč/h
Celkově	20 000 Kč

Tab. 2: Zhodnocení

4.14 Přínos práce

Výše navrhované řešení aplikace je uskutečněno podle požadavků klienta Gaynet. Klient požadoval aplikaci pro platformu iOS, která by nahradila přístup na stránky seznamky na mobilních zařízeních. Aplikace plně nahrazuje webové stránky a uživatel se tak může stejně jako na webu registrovat nebo přihlásit ke svému účtu. Může prohlížet ostatní uživatele, hodnotit jejich fotky, psát si zprávy a zobrazovat uživatele v okolí.

Nová moderní aplikace může mít pozitivní vliv na nové potenciační uživatele. Je zde příležitost, že se uživatelská základna rozroste. Webové stránky nejsou příliš vyhovující na dnešní standarty, a tak by měla tato aplikace usnadnit příliv nových uživatelů nebo dostat zpátky uživatele, kteří přešli ke konkurenci.

Závěr

Cílem práce bylo navrhnout a vytvořit aplikaci. Jako aplikaci jsem si zvolil vytvoření seznamky, která by plně nahradila webovou službu.

Nejprve jsme provedli analýzu trhu, ve které jsme zjistili, že i když má Android největší podíl na trhu, zisky z App Storu jsou stále vyšší. Poté jsem analyzoval monetizační metody, které lze uplatnit a podle typu aplikace bylo rozhodnuto, že nejlepším řešením je umožnit určitý obsah zdarma a veškerý obsah umožnit pouze prémiovým uživatelům. Dále jsem provedl analýzu stávající webové služby a vytknul některé věci, které web postrádá pro mobilní zařízení a kvůli kterým se vyvíjí tato mobilní aplikace.

V další části jsem naplnil stanovený cíl. Nejdříve jsem navrhl aplikaci, vytvořil digramy, které zobrazovaly, jak by měla aplikace fungovat a které obrazovky budou zapotřebí. Poté jsem vytvořil jednotlivé obrazovky, které zastupovaly jednotlivé služby webu. Veškerý kód byl napsán v jazyce Swift 3. Aplikace byla napsaná v nejnovějším XCode 8.2 a používá moderní knihovny, především Alamofire a SwiftyJSON, které se starají o posílání požadavků. Aplikace je umístěna na App Store, ale zatím pouze pro testování. Přístup k ní mají pouze vývojáři a klient, aby mohl vznést různé připomínky a dodatky před tím, než aplikace bude spuštěna naživo.

Naprogramování výše popsané aplikace bylo pro mě velkým přínosem. Jednalo se o první větší projekt, na kterém jsem pracoval. Získal jsem mnoho zkušeností, které mohu uplatnit v dalších projektech a významnou referenci do svého portfolia.

Seznam dostupných zdrojů

- [1] WIGMORE, Ivy. Mobile app. WhatIs.com [online]. 2013 [cit. 2017-03-03]. Dostupné z: <http://whatIs.techtarget.com/definition/mobile-app>
- [2] Informace o mobilních platformách. Sun Marketing [online]. [cit. 2017-03-03]. Dostupné z: <http://www.sunmarketing.cz/mobilni-aplikace/informace-o-mobilnich-platformach>
- [3] Mobilní platformy. Pixelspace [online]. [cit. 2017-03-03]. Dostupné z: <http://vyvojmobilnichaplikaci.com/mobilni-platformy>
- [4] Swift. Developer Apple [online]. [cit. 2017-03-03]. Dostupné z: <https://developer.apple.com/swift>
- [5] About Swift. Developer Apple [online]. [cit. 2017-03-03]. Dostupné z: https://developer.apple.com/library/ios/documentation/Swift/Conceptual/Swift_Programming_Language
- [6] THOMAS, Kaya. Getting Started with iOS: Objective - C vs. Swift. In: CodeNewbie [online]. 2015 [cit. 2017-03-03]. Dostupné z: <http://www.codenewbie.org/blogs/getting-started-with-ios-objective-c-vs-swift>
- [7] BEAL, Vangie. Apple App Store. In: Webopedia [online]. [cit. 2017-03-03]. Dostupné z: http://www.webopedia.com/TERM/A/apple_app_store.html
- [8] Developer Apple [online]. [cit. 2017-03-03]. Dostupné z: <https://developer.apple.com>
- [9] Smartphone OS Market Share, 2015 Q2. IDC [online]. [cit. 2017-03-03]. Dostupné z: <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>

- [10] CocoaPods [online]. [cit. 2017-03-15]. Dostupné z: <https://cocoapods.org/>
- [11] Alamofire [online]. [cit. 2017-03-15]. Dostupné z: <https://github.com/Alamofire/Alamofire>
- [12] SwiftyJSON [online]. [cit. 2017-03-15]. Dostupné z: <https://github.com/SwiftyJSON/SwiftyJSON>
- [13] SWRevealViewController [online]. [cit. 2017-03-15]. Dostupné z: <https://github.com/John-Lluch/SWRevealViewController>
- [14] BRYXBanner [online]. [cit. 2017-03-15]. Dostupné z: <https://github.com/bryx-inc/BRYXBanner>
- [15] SWOT Analýza [online]. [cit. 2017-03-15]. Dostupné z: <https://managementmania.com/cs/swot-analyza>
- [16] SWOT Analýza [online]. [cit. 2017-03-15]. Dostupné z: <http://excel-navod.fotopulos.net/swot-analyza.html>
- [17] ČÁPKA, David. Úvod do objektově orientovaného programování v C# [online]. [cit. 2017-03-15]. Dostupné z: <http://www.itnetwork.cz/csharp/oop/c-sharp-tutorial-uvod-do-objektove-orientovaneho-programovani>
- [18] Cumulative number of apps downloaded from the Google Play as of May 2016 (in billions). *Statista* [online]. [cit. 2017-01-22]. Dostupné z: <https://www.statista.com/statistics/281106/number-of-android-app-downloads-from-google-play/>

- [19] Smartphone OS Market Share, 2016 Q3. *IDC* [online]. [cit. 2017-01-22].
Dostupné z:
<http://www.idc.com/promo/smartphone-market-share/os;jsessionid=8701DD11220DBE1CE835D694826118AF>
- [20] App Store Statistics Roundup. *Business of Apps* [online]. [cit. 2017-01-22].
Dostupné z: <http://www.businessofapps.com/app-store-statistics-roundup/>
- [21] Get Started with Publishing. *Developer Android* [online]. [cit. 2017-01-22].
Dostupné z: <https://developer.android.com/distribute/googleplay/start.html>
- [22] Most popular Apple App Store categories in December 2016, by share of available apps. *Statista* [online]. [cit. 2017-01-22]. Dostupné z:
<https://www.statista.com/statistics/270291/popular-categories-in-the-app-store/>
- [23] Apple Developer Program. *Developer Apple* [online]. [cit. 2017-01-22].
Dostupné z: <https://developer.apple.com/programs/how-it-works/>
- [24] 2016 recap: 90% of Google Play's revenue came from games (and more fun stats!). *Android Authority* [online]. [cit. 2017-01-22]. Dostupné z:
<http://www.androidauthority.com/2016-recap-90-percent-google-play-revenue-gaming-fun-stats-743626/>
- [25] China overtakes the U.S. in iOS App Store revenue. *Tech Crunch* [online]. [cit. 2017-01-22]. Dostupné z: <https://techcrunch.com/2016/10/20/china-overtakes-the-u-s-in-ios-app-store-revenue/>
- [26] Gaynet [online]. [cit. 2017-05-15]. Dostupné z: <http://www.gaynet.ch/>
- [27] Apptentive [online]. [cit. 2017-05-15]. Dostupné z:
<https://www.apptentive.com/blog/2016/09/20/6-mobile-app-monetization-strategies/>

Seznam obrázků

Obr. 1: Komplexní řazení objektů [6]	14
Obr. 2: SWOT analýza [16]	16
Obr. 3: Rozdělení světového trhu podle OS [19]	18
Obr. 4: Top 3 Nejvíce utrácějící země na App Store [24].....	19
Obr. 5: Gaynet [26]	23
Obr. 6: Diagram přihlašování.....	28
Obr. 7: Hlavní diagram.....	29
Obr. 8: Návrh tříd.....	30
Obr. 9: Přihlašování	31
Obr. 10: Registrace.....	32
Obr. 11: Uživatelé v okolí.....	33
Obr. 12: Mapa uživatelů v okolí.....	34
Obr. 13: Filter.....	35
Obr. 14: Profil uživatele	36
Obr. 15: Hodnocení fotek.....	37
Obr. 16: Menu	38
Obr. 17: Konverzace	39
Obr. 18: Zprávy	40
Obr. 19: Moje fotky a videa	41
Obr. 20: Listy	42
Obr. 21: Nastavení	43

Seznam tabulek

Tab. 1: SWOT analýza	24
Tab. 2: Zhodnocení	44